COLUMN GENERATION

OR Software Tools, IAP 2015

Chiwei Yan, MIT Operations Research Center

January 29th, 2015

Background

- *Column Generation* is a solution approach to tackle large LPs with lots of variables. (in contrast to *cutting planes*)
- *Column Generation* is also about a modeling paradigm called *column-wise* modeling.
- Many real-world optimization problems involve complex and tedious constraints which are hard to formulate through constraint-wise approach:
  - *Amazon Fresh* wants to route n vehicles to serve m customers' grocery demand everyday. Each customer has a specific time window for delivery, each vehicle has a maximum distance to travel due to fuel constraints.
  - *Alaska Airlines* wants to route n aircraft to cover m flights. Each route is a sequence of flights flew by the same aircraft. Aircraft can only be routed through connectable flights (flight 1's destination=flight 2's origin, flight 2's departure time is later than flight 1's arrival time).

In this lecture, you will learn…

- Column-wise modeling approach:
  - Transform *“Complex Constraints”* into *“Definition of Variables”*
- Column generation solution approach:
  - The above transformation is usually accomplished at the cost of huge increasing in number of decision variables
  - (Delayed) column generation helps to solve the program by generating only a small subset of the variables
- Implementing column generation and solving *Alaska Airlines’ aircraft routing problem* in Julia/JuMP.
- (Hopefully) introduce you a new way to formulate optimization problems in your own research projects.

Column Generation Essentials

- Consider the following standard form LP.

$$\begin{aligned} Min \quad & c^T x \\ s.t. \quad & Ax = b \\ & x \geq 0 \end{aligned} \quad \Longleftrightarrow \quad \begin{aligned} Min \quad & c_1 x_1 + c_2 x_2 + \cdots + c_n x_n \\ s.t. \quad & A_1 x_1 + A_2 x_2 + \cdots + A_n x_n = b \\ & x_1, x_2, \cdots, x_n \geq 0 \end{aligned}$$

- Sufficient condition for basis B to be optimal
  - Reduced cost of any variable $x_k = c_k - c_B^T B^{-1} A_k = c_k - r^T A_k \geq 0$
  - r is the dual prices associated with basis B
- Start the solution process with a subset of variables, WLOG, only x_1 and x_2. Get the optimal dual solution r.

(Restricted Master Problem)
$$\begin{aligned} Min \quad & c_1 x_1 + c_2 x_2 \\ s.t. \quad & A_1 x_1 + A_2 x_2 = b \qquad (r) \\ & x_1, x_2 \geq 0 \end{aligned}$$

- Solve the following sub-problem. If $z^* < 0$, then add optimal column A_{k^*} to the restricted master problem. Resolve it. If $z^* \geq 0$, the current solution from restricted master problem is optimal.

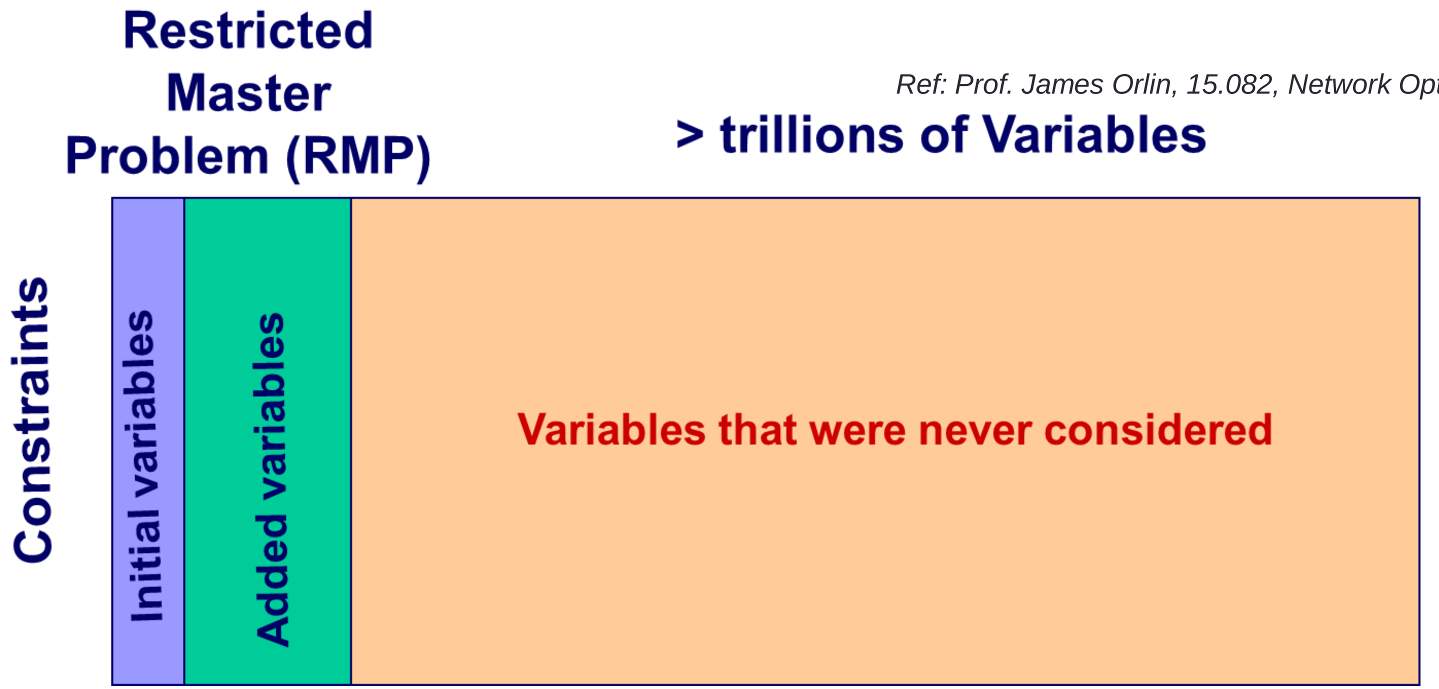
(Sub-problem)
$$z^* = Min_{k \in remaining\ variables} \quad c_k - r^T A_k$$

Column Generation Essentials

- The beauty of column generation is the sub-problem in certain cases are straightforward to model and solve, *than it appears to be.*☺

(Sub-problem) $$z^* = Min_{k \in remaining\ variables} \quad c_k - r^T A_k \begin{cases} \text{shortest path problem} \\ \text{knapsack problem} \\ \dots \end{cases}$$

- Column Generation in one figure:



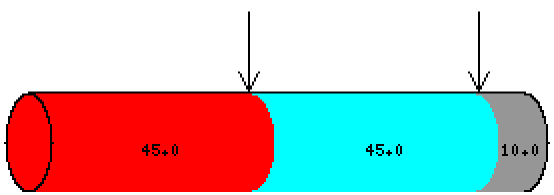
Ref: Prof. James Orlin, 15.082, Network Optimization

Why we care about problems with lots of variables?

- **Many problems are easy to model column-wisely, though at the cost of increasing number of variables.**

Warm-Up Example: Cutting Stock Problem

- Given paper rolls of fixed width (100 inches) and a set of orders for rolls of smaller widths (14, 31, 36, 45 inches), the objective of the Cutting Stock Problem is to determine how to cut the 100-inch rolls into smaller widths to fulfill the orders in such a way as to minimize the number of 100-inch paper rolls used.



Order Width	Quantity Ordered
14	211
31	395
36	610
45	97

Ref: *http://www.neos-guide.org/content/cutting-stock-problem*

Define n_1, n_2, n_3, n_4 as the number of cuts for each order width for a paper roll,

$$14n_1 + 31n_2 + 36n_3 + 45n_4 \leq 100$$

n_1, n_2, n_3, n_4 are integers

However, each paper roll may have different cutting patterns, $n_{1t}, n_{2t}, n_{3t}, n_{4t}$ as the number of cuts for each order width for paper roll t……………………

How to formulate the problem?

Answer: ***Column-wise modeling:***
Bring constraints into definition of variables.

Warm-Up Example: Cutting Stock Problem

- Sets
  - I, set of order widths; J, set of feasible cutting patterns $= (a_{1j}, a_{2j}, a_{3j}, a_{4j})$
- Parameters
  - a_{ij}, is the number of cuts of order width i in cutting pattern j, $\forall i \in I, j \in J$
  - b_i, is the order quantity of order width i
- Decision Variables
  - x_j, is the number of paper rolls using cutting pattern j, $\forall i \in I, j \in J$

$$
\begin{aligned}
MIN \quad & \sum_{j \in J} x_j \\
s.t. \quad & \sum_{j \in J} a_{ij} x_j \geq b_i, && \forall i \in I \\
& x_j \; is \; integer, && \forall j \in J
\end{aligned}
$$

$|J|$ equals to number of different solutions for

$$14a_{1j} + 31a_{2j} + 36a_{3j} + 45a_{4j} \leq 100$$

$a_{1j}, a_{2j}, a_{3j}, a_{4j}$ are integers

Difficulty: $|J|$ is large. Hard to enumerate $x_j, \forall j \in J$ and include them into the model…

Solution: Column Generation

Warm-Up Example: Cutting Stock Problem

- Solve the LP relaxation using column generation:

STEP 1:
- Initialize the restricted master problem with two patterns
  - width 14: 1 cut; width 31: 1 cut; width 36: 0 cut; width 45: 1 cut (14+31+45=90<100)
  - width 14: 0 cut; width 31: 0 cut; width 36: 2 cut; width 45: 0 cut (36*2=72<100)
  - Why those two patterns?

restricted master problem:

$$MIN \quad x_1 + x_2$$

$$s.t. \quad \begin{pmatrix}1\\1\\0\\1\end{pmatrix} x_1 + \begin{pmatrix}0\\0\\2\\0\end{pmatrix} x_2 \geq \begin{pmatrix}211\\395\\610\\97\end{pmatrix} \begin{array}{l} r_1\\ r_2\\ r_3\\ r_4\end{array}$$

$$x_1, x_2 \geq 0$$

STEP 2:
- Solve the restricted master problem
  - Optimal dual solution: (r_1, r_2, r_3, r_4)

STEP 3:
- Recall the sufficient optimal condition for LP
  - All variables' reduced costs ≥ 0
  - Reduced cost of a potential variable x_k

$$rc(x_k) = 1 - (r_1\ r_2\ r_3\ r_4)\begin{pmatrix}a_{1k}\\a_{2k}\\a_{3k}\\a_{4k}\end{pmatrix}$$

sub-problem (knapsack problem):

$$z^* = MAX \quad \sum_{i=1}^{4} r_i a_{ik}$$

$$s.t. \quad 14a_{i1} + 31a_{i2} + 36a_{i3} + 45a_{i4} \leq 100$$

$$a_{i1}, a_{i2}, a_{i3}, a_{i4} \ are\ integers$$

Order Width	Quantity Ordered
14	211
31	395
36	610
45	97

STEP 4:
- If $z^* \leq 1$, the optimal solution of restricted master problem is optimal for the original LP relaxation.

STEP 5:
- If $z^* > 1$, add optimal solution $(a_{1k}, a_{2k}, a_{3k}, a_{4k})$ from sub-problem as a new column (variable) to restricted master problem. GOTO Step 2.

Warm-Up Example: Cutting Stock Problem

- Solve the LP relaxation using column genera

STEP 1: • Initialize the restricted master problem w
  - width 14: 1 cut; width 31: 1 cut; width 36: 0 (14+31+45=90<100)
  - width 14: 0 cut; width 31: 0 cut; width 36: 2 (36*2=72<100)
  - Why those two patterns?

restricted master problem, after one column added:

$$MIN \quad x_1 + x_2 + x_k$$

$$s.t. \quad \begin{pmatrix}1\\1\\0\\1\end{pmatrix} x_1 + \begin{pmatrix}0\\0\\2\\0\end{pmatrix} x_2 + \begin{pmatrix}a_{1k}\\a_{2k}\\a_{3k}\\a_{4k}\end{pmatrix} x_k \geq \begin{pmatrix}211\\395\\610\\97\end{pmatrix}$$

$$x_1, x_2, x_k \geq 0$$

STEP 2: • Solve the restricted master problem
  - Optimal dual solution: (r_1, r_2, r_3, r_4)

STEP 3: • Recall the sufficient optimal condition for LP
  - All variables' reduced costs ≥ 0
  - Reduced cost of a potential variable x_k

$$rc(x_k) = 1 - (r_1\ r_2\ r_3\ r_4)\begin{pmatrix}a_{1k}\\a_{2k}\\a_{3k}\\a_{4k}\end{pmatrix}$$

sub-problem (knapsack problem):

$$z^* = MAX \quad \sum_{i=1}^{4} r_i a_{ik}$$

$$s.t. \quad 14a_{i1} + 31a_{i2} + 36a_{i3} + 45a_{i4} \leq 100$$

$$a_{i1}, a_{i2}, a_{i3}, a_{i4} \ are\ integers$$

Order Width	Quantity Ordered
14	211
31	395
36	610
45	97

STEP 4: • If $z^* \leq 1$, the optimal solution of restricted master problem is optimal for the original LP relaxation.

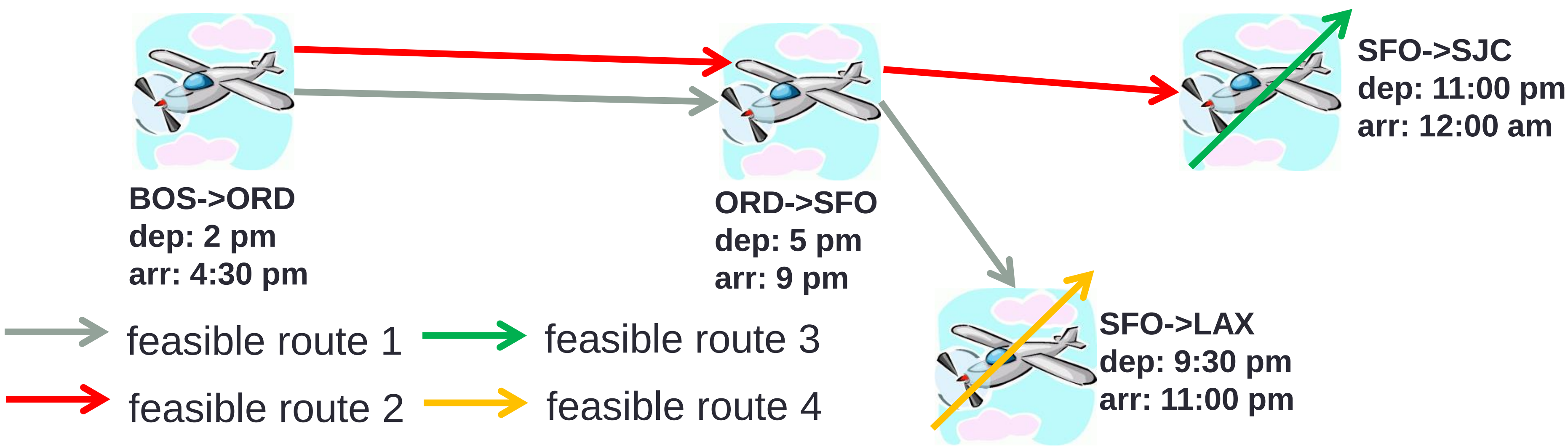
STEP 5: • If $z^* > 1$, add optimal solution $(a_{1k}, a_{2k}, a_{3k}, a_{4k})$ from sub-problem as a new column (variable) to restricted master problem. GOTO Step 2.

Warm-Up Example: Cutting Stock Problem

- Let’s now implement and solve this cutting stock problem in Julia/JuMP!

CG is much more powerful than just cutting paper rolls…

- *Alaska Airlines* wants to route n aircraft to cover m flights. Each route is a sequence of flights flew by the same aircraft. Aircraft can only be routed through connectable flights (flight 1's destination=flight 2's origin, flight 2's departure time is later than flight 1's arrival time).
- ***How to model the flight connection constraints?***
- **Solution:**
  - Forget about complex constraints first, bring them to the definition of variable!
    - Set: R, set of feasible route can be flew by an aircraft (connectable)
    - $x_r \in \{0,1\}, r \in R$, equals 1 if feasible route r is used; 0, otherwise.

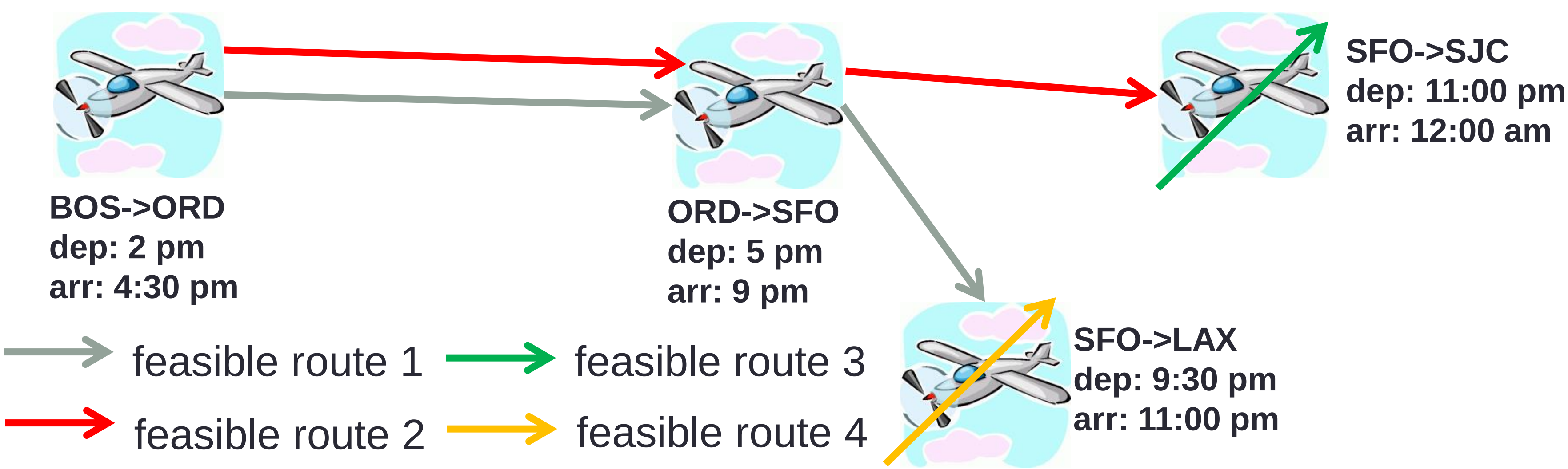


Aircraft Routing Problem

- In order to ensure each flight is covered by exactly one route

- $$A = \begin{array}{ccccl} & & & & \\ \left(\begin{matrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{matrix}\right) & \begin{matrix} \text{BOS->ORD} \\ \text{ORD->SFO} \\ \text{SFO->LAX} \\ \text{SFO->SJC} \end{matrix} \\ \begin{matrix} \text{Route 1} & \text{Route 2} & \text{Route 3} & \text{Route 4} \end{matrix} & \end{array}, \quad \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \end{pmatrix} x_1 + \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \end{pmatrix} x_2 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} x_3 + \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} x_4 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

$$Ax = 1$$

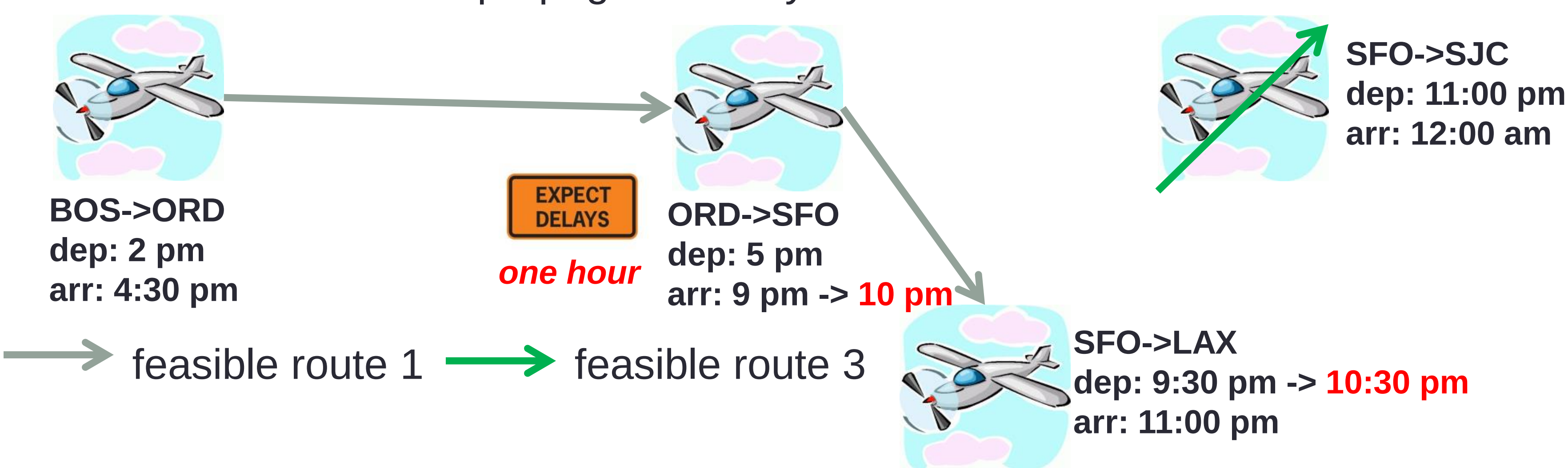


Aircraft Routing Problem

- In order to ensure each flight is covered by exactly one route
  - $A = \underbrace{\begin{pmatrix} 0 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 0 \end{pmatrix}}_{\text{\# of feasible routes}=|R|} \left.\vphantom{\begin{pmatrix} 0 \\ \vdots \\ 1 \end{pmatrix}}\right\}$ # of flights=m, $A_{ij} = 1$ if route j covers flight i
  - $Ax = 1 \quad \Longleftrightarrow \quad A_1 x_1 + A_2 x_2 + \cdots + A_{|R|} x_{|R|} = 1$
- In order to ensure at most n aircraft can be used
  - $x_1 + x_2 + \cdots + x_{|R|} \leq n$
- Binary: $x_1, x_2, \cdots, x_{|R|} \in \{0,1\}$

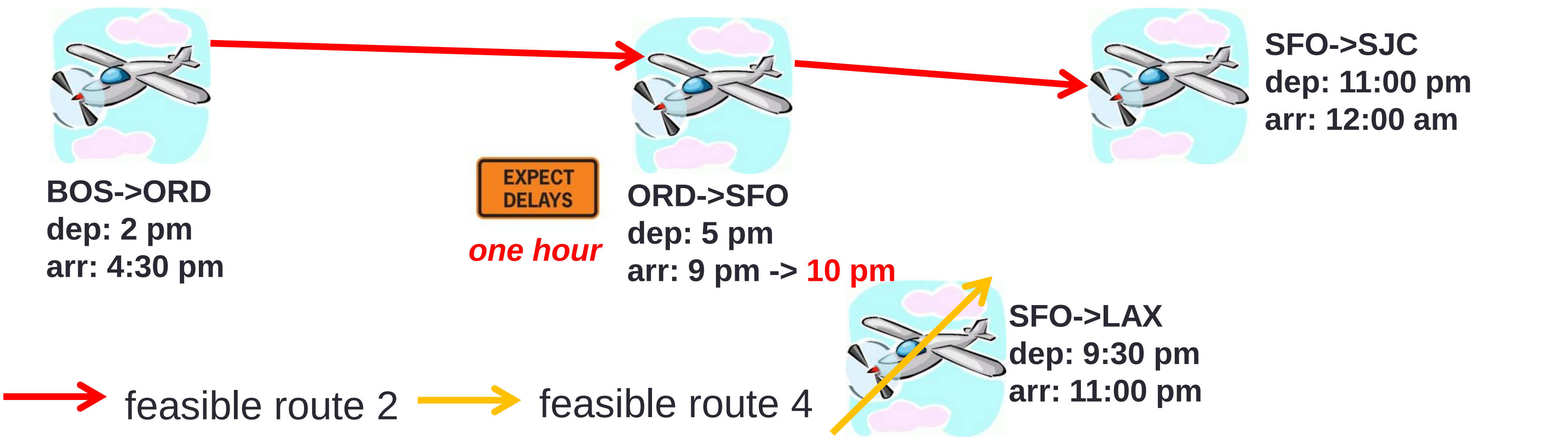
Aircraft Routing Problem

- Objective: Minimize *Propagated Delay*
  - Each flight has an arrival delay
  - Two connecting flight flew by same aircraft has a 30-minute minimum separation time
  - Flight ORD->SFO has 60 minutes arrival delay
  - Using route 1 and route 3
    - Route 1 has one hour propagated delay on flight SFO->LAX
    - Route 3 has 0 propagated delay



Aircraft Routing Problem

- Objective: Minimize *Propagated Delay*
  - Each flight has an arrival delay
  - Two connecting flight flew by same aircraft has a 30-minute minimum separation time
  - Flight ORD->SFO has 60 minutes arrival delay
  - Using route 2 and route 4
    - Route 2 has 0 propagated delay
    - Route 4 has 0 propagated delay



Aircraft Routing Problem

- Set
  - R, set of feasible route can be flew by an aircraft (connectable)
- Decision Variables
  - $x_r \in \{0,1\}, r \in R$, equals 1 if feasible route r is used; 0, otherwise.
- Parameter
  - m, number of flights; n, number of aircraft
  - $A \in \{0,1\}^{m\times|R|}, A_{ij} = 1$ if $route\ j$ covers $flight\ i$; 0, otherwise
  - $p_r, r \in R$, total propagated delay on route r
- Formulation:

$$
\begin{aligned}
Min \quad & \sum_{r\in R} p_r x_r \\
s.t. \quad & Ax = 1 \\
& \sum_{r\in R} x_r \le n \\
& x_r \in \{0,1\}, \forall r \in R
\end{aligned}
$$

Huge number of variables

⇩

Column Generation!

Aircraft Routing Problem - CG

- Reduced cost of a potential route variable x_r
  - $p_r - s^T A_r - t$
  - $A_r[j] = 1$ if flight j is covered by route r; 0, otherwise.

$$\text{Min} \sum_{r \in R} p_r x_r$$
$$s.t. \quad A_1 x_1 + \cdots A_{|R|} x_{|R|} = 1 \quad (s)$$
$$\sum_{r \in R} x_r \leq n \quad (t)$$
$$0 \leq x_r \leq 1, \forall r \in R$$

$$p_r - s^T A_r - t = p_r - \sum_{f \in r} s_f - t,$$

$f \in r$ are the flights route r covers

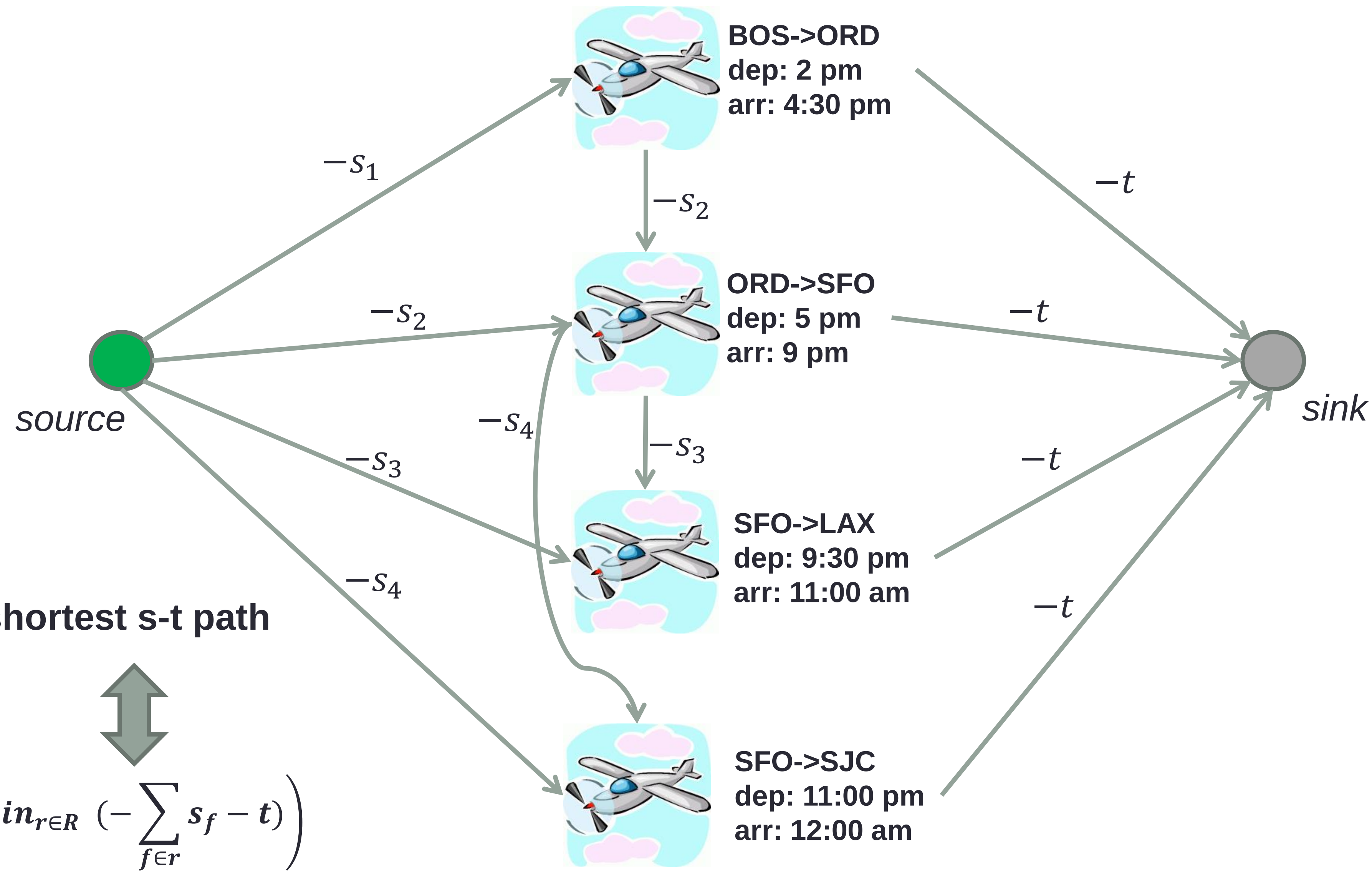
(sub-problem) $z^* = Min_{r \in R} \ (p_r - \sum_{f \in r} s_f - t) \ < 0\,??$

Heuristic

A shortest path problem!

(sub-problem) $z^* = \boxed{\left(Min_{r \in R} \ (-\sum_{f \in r} s_f - t)\right)} + p_r \ < 0\,??$

Aircraft Routing Problem - CG

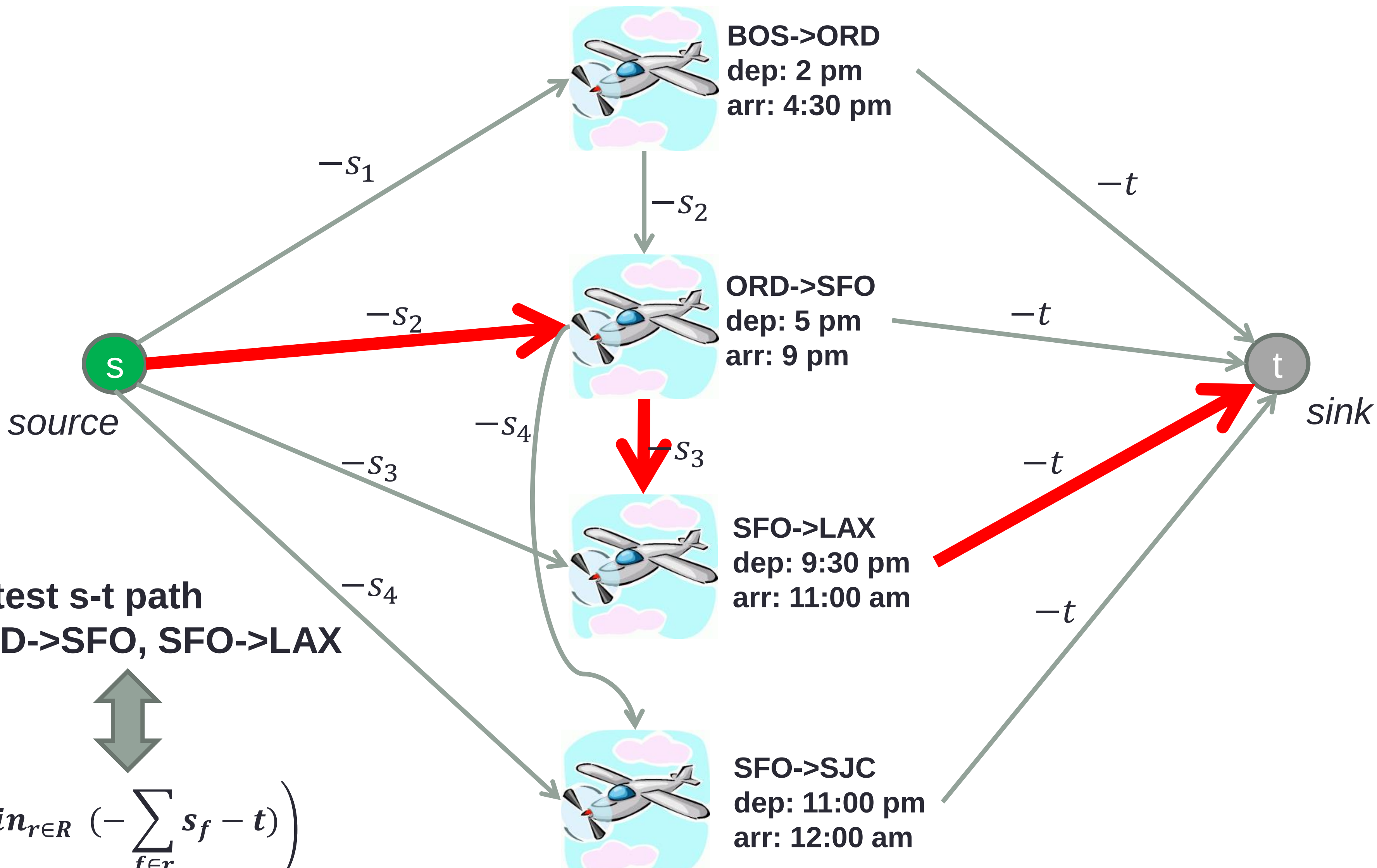


shortest s-t path

$$\Updownarrow$$

$$\left(\boldsymbol{Min}_{r \in R} \; \left(-\sum_{f \in r} s_f - t\right) \right)$$

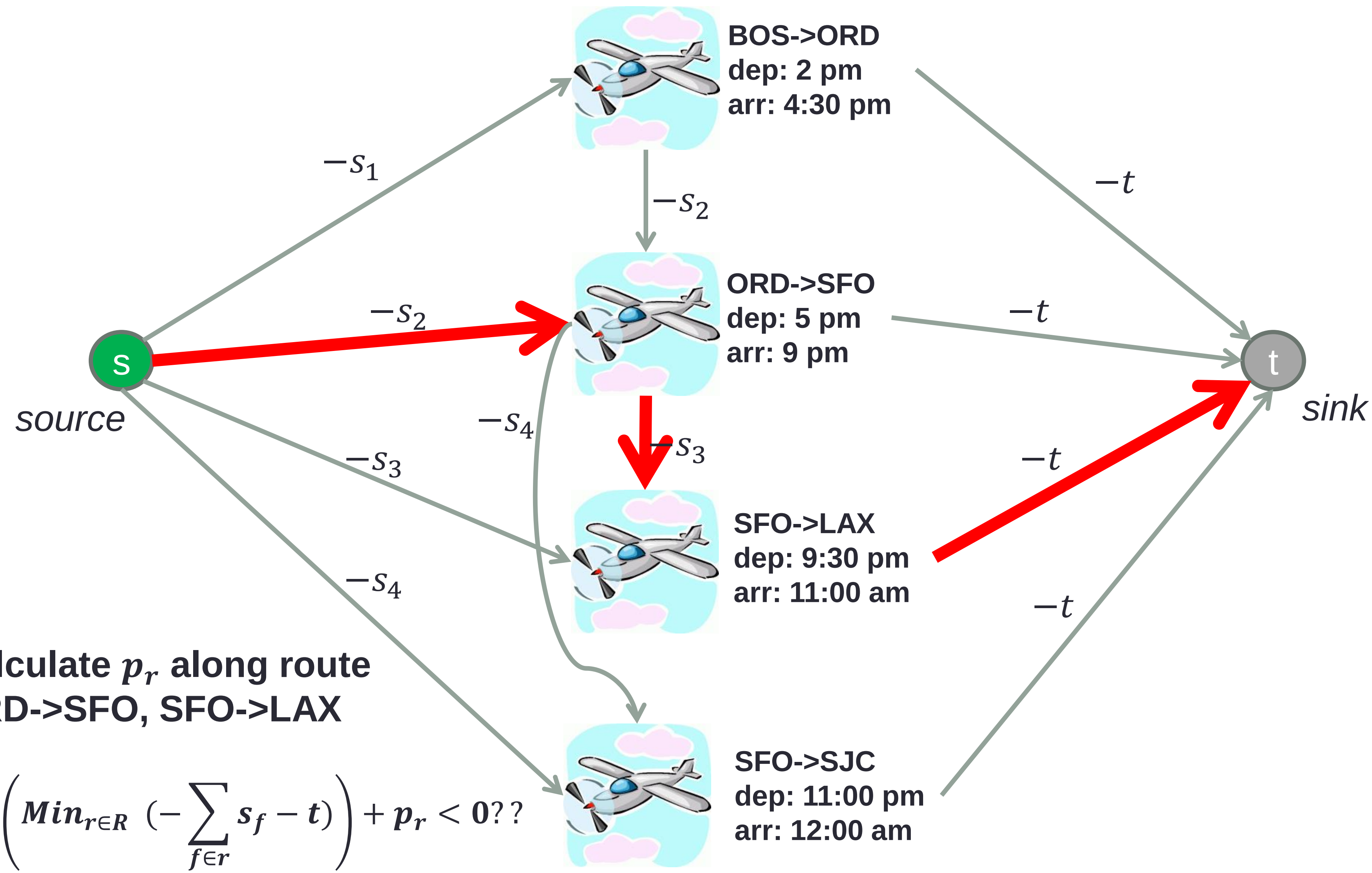
Aircraft Routing Problem - CG



shortest s-t path
= ORD->SFO, SFO->LAX

$$\left(\boldsymbol{Min}_{r\in R}\ \left(-\sum_{f\in r} s_f - t\right) \right)$$

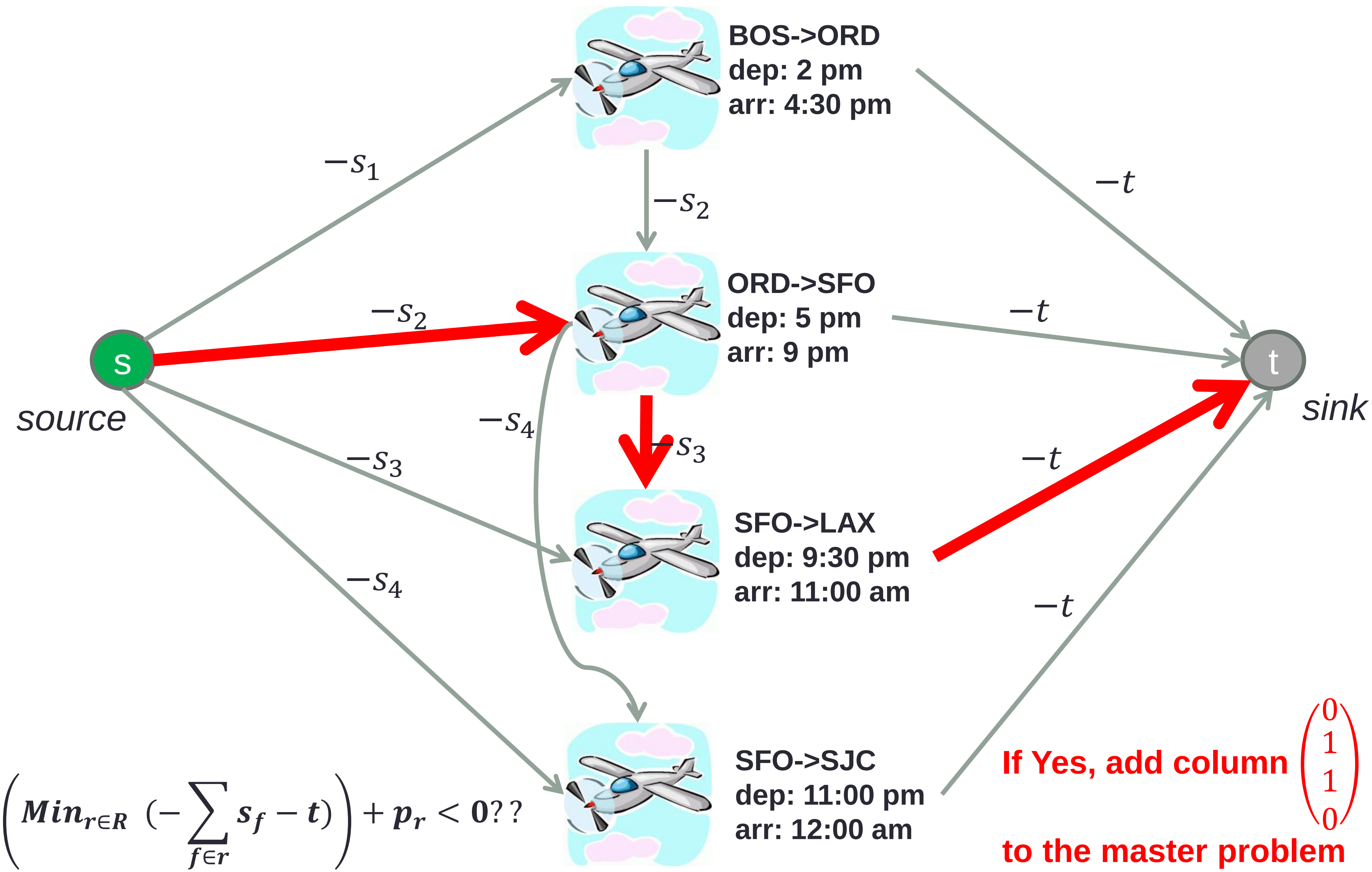
Aircraft Routing Problem - CG



Calculate p_r along route ORD->SFO, SFO->LAX

$$z^* = \left(Min_{r \in R} \; (-\sum_{f \in r} s_f - t) \right) + p_r < 0??$$

Aircraft Routing Problem - CG



$$z^* = \left(Min_{r \in R} \ (-\sum_{f \in r} s_f - t) \right) + p_r < 0??$$

Aircraft Routing Problem - CG

- Let’s now implement and solve this aircraft routing problem in Julia/JuMP!

More on column-wise modeling

- What if American Airlines has some additional constraints?
  - An aircraft cannot fly more than 18 hours consecutively?
  - An aircraft must return to either ORD or SFO for night-time maintenance?
- Can you now try to model *Amazon Fresh's* vehicle routing problem using column-wise modeling?
  - *Amazon Fresh* wants to route n vehicles to serve m customers' grocery demand everyday. Each customer has a specific time window for delivery, each vehicle has a maximum distance to travel due to fuel constraints.

Summary

- We introduce column-wise modeling for optimization problems. For certain types of problems, this approaches greatly simplifies the modeling efforts however increases decision variable dramatically.
- Column generation solution approach comes to rescue by generating only a small subset of variables.

More on column generation

- Speeding up…
  - Generating multiple columns at one iteration
  - Stabilized dual solution
  - Column management
- Column generation on integer programs
  - Branch-and-Price
- SCIP Optimization Suite, http://scip.zib.de/
- Try column-wise modeling and using CG in your future research.
- Questions? chiwei@mit.edu